

# Styling default JavaFX Dialogs

Asked 8 years, 7 months ago   Modified 6 years, 8 months ago   Viewed 44k times



35



I'm looking for a way to style the default JavaFX Dialog ( `javafx.scene.control.Dialog` ).

I tried to get the `DialogPane` and add a stylesheet, but it covers only a small piece of the dialog. I would prefer to style only with an external css file and without to add `styleClasses` over the code. This would look messy (header, content, own content on the content and more..)

I googled already alot and only found examples for `ControlsFX`, but since `jdk8_40` JavaFX has it's own Dialogs i use them now.

Any suggestions?

## Edit:

Since José Pereda posted the solution i created my own `dialog.css`. I'll post it here because it covers the whole dialog and maybe someone want's to copy&paste it. Note `.dialog-pane` is already a given `styleClass` name so you don't need to apply your own. Of course, Josés is more fine detailed.

```
.dialog-pane {
    -fx-background-color: black;
}

.dialog-pane .label {
    -fx-text-fill: white;
}

.dialog-pane:header .header-panel {
    -fx-background-color: black;
}

.dialog-pane:header .header-panel .label {
    -fx-font-style: italic;
    -fx-font-size: 2em;
}
```

[javafx](#)   [dialog](#)   [javafx-8](#)

Share   Improve this question   Follow

edited Dec 22, 2016 at 16:21

asked Feb 9, 2015 at 18:53



xoned

2,701

2

31

43

Sorted by:

1 Answer

Highest score (default)



71



You can style your dialogs with your own css file, but for that you need to take into consideration that the dialog is in fact a new stage, with a new scene, and the root node is a `DialogPane` instance.

So once you create some dialog instance:



```
@Override
public void start(Stage primaryStage) {
    Alert alert = new Alert(AlertType.CONFIRMATION);
    alert.setTitle("Confirmation Dialog");
    alert.setHeaderText("This is a Custom Confirmation Dialog");
    alert.setContentText("We override the style classes of the dialog");
    ...
}
```

you can access to its dialog pane and add your own style sheet and your own class selector:

```
DialogPane dialogPane = alert.getDialogPane();
dialogPane.getStylesheets().add(
    getClass().getResource("myDialogs.css").toExternalForm());
dialogPane.setStyleClass().add("myDialog");
```

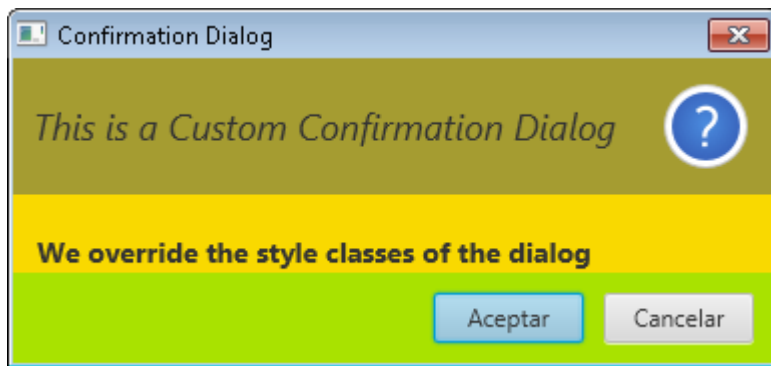
Now the trick is knowing all the rules a `Dialog` style sheet has implemented by default.

And that's a difficult task... since they are not in the `modena.css` file, as for all the regular controls. On the contrary, they are found in the `modena.bss` file, a binary file located in the `jfxrt.jar` under private packages.

After some digging I've managed to get those rules, so your custom `myDialogs.css` file will look something like this:

```
.myDialog{
    -fx-background-color: #f9d900;
}
.myDialog > *.button-bar > *.container{
    -fx-background-color: #a9e200;
}
.myDialog > *.label.content{
    -fx-font-size: 14px;
    -fx-font-weight: bold;
}
.myDialog:header *.header-panel{
    -fx-background-color: #a59c31;
}
.myDialog:header *.header-panel *.label{
    -fx-font-size: 18px;
    -fx-font-style: italic;
    -fx-fill: #292929;
}
```

And you will have your styled dialog:



Note that being a `bss` file under private packages, these selectors can change without notice in future releases.

## EDIT

I've just found that the `.dialog-pane` selector is already part of [modena.css](#) in the last 8u40 [early versions](#), so you can find all the selectors and rules applied to the dialog pane there.

Share Improve this answer Follow

edited Feb 10, 2015 at 19:24

answered Feb 9, 2015 at 23:08



**José Pereda**

**44.3k** 7 104 132

- 1 I've just edited my answer, as `modena.css` already includes the `.dialog-pane` selector in the latest 8u40 ea. My other post was related to [openjfx-dialogs](#), the inbetween project extracted from ControlsFX, valid for 8u20, 8u25 and 8u31 versions. – José Pereda Feb 10, 2015 at 19:29

Thanks for this answer, it was very helpful! However, I have noticed that using `:header` is not working anymore, so I just used `.myDialog .header-panel`, which works perfectly. I am also using 8u40.. – bashoogzaad Mar 12, 2015 at 11:42